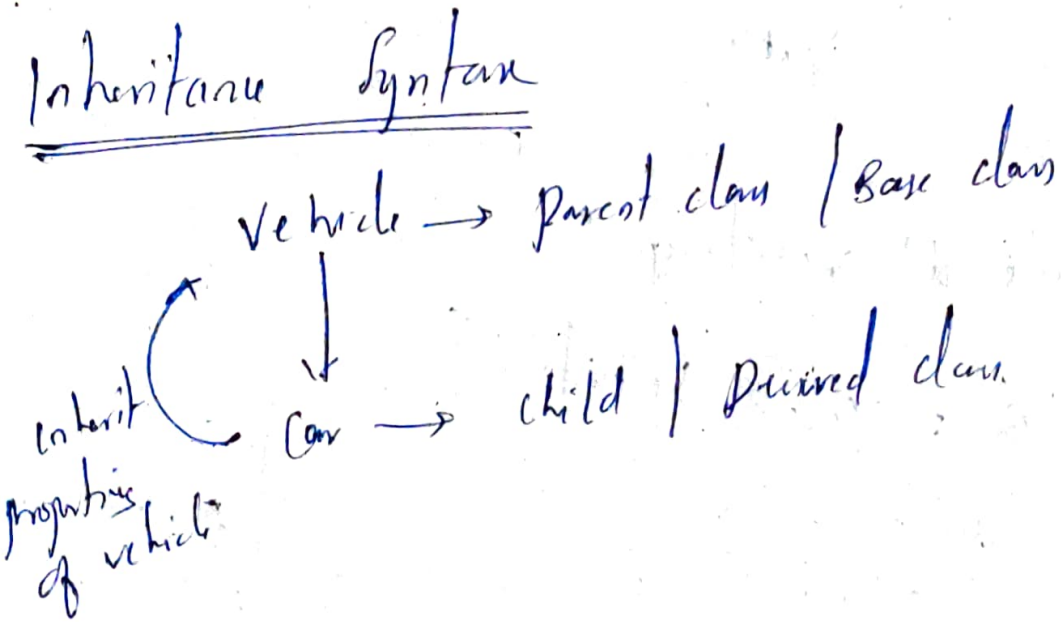
3) Inheritance Syntax



.py

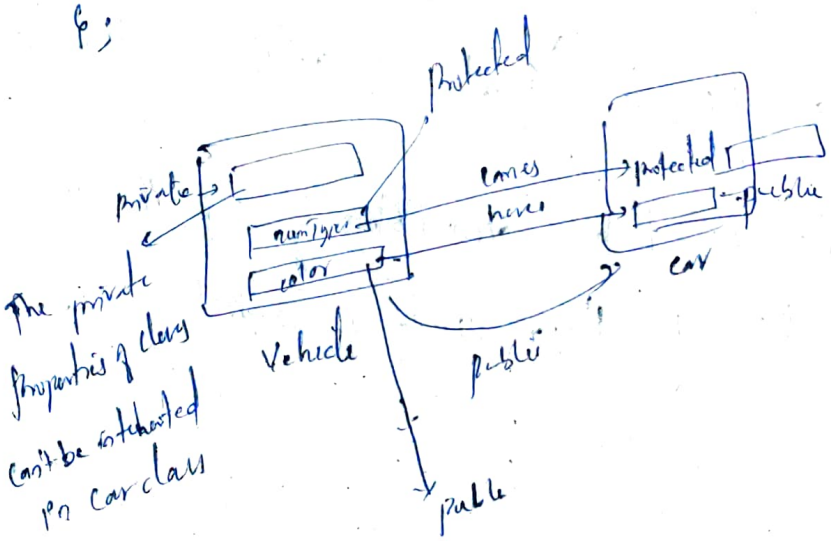
1) Vehicle.cpp

```
class Vehicle {
private:
    int maxSpeed;

protected:
    int numTyres;

public:
    string color;

};
```



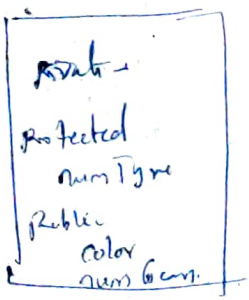
2) Car.cpp

```
#include "vehicle.cpp"

class Car : public Vehicle {

public:
    int numGears;

void print() {
    cout << numTyres << endl;
    cout << color << endl;
    cout << numGears << endl;
}
```



Base class		Child class
public	becomes →	public
protected	becomes →	protected
private	→	X

Vehicle.cpp

Vehicle V;

V.C:

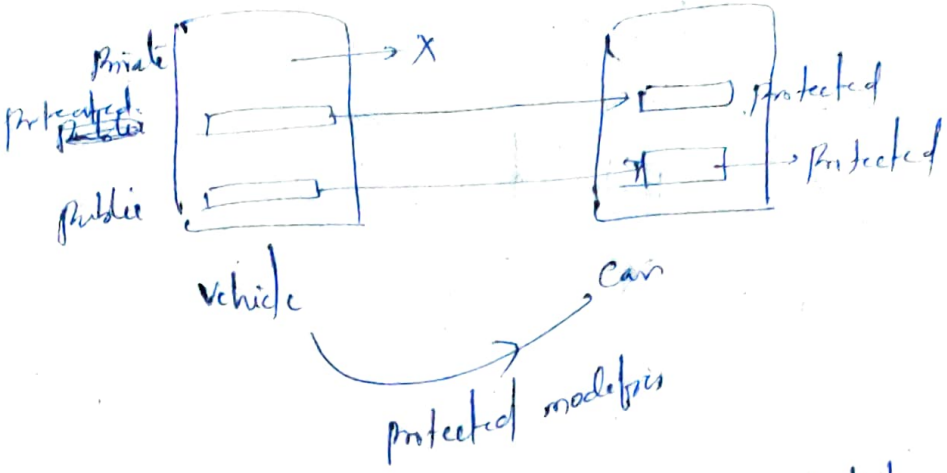
```
#include <iostream>
using namespace std;
#include "vehicle.cpp"
#include "Car.cpp"

int main() {
    vehicle v;
    v.color = "blue";
    // v.maxspeed = 100;
    // v.numTyres = 4;
```

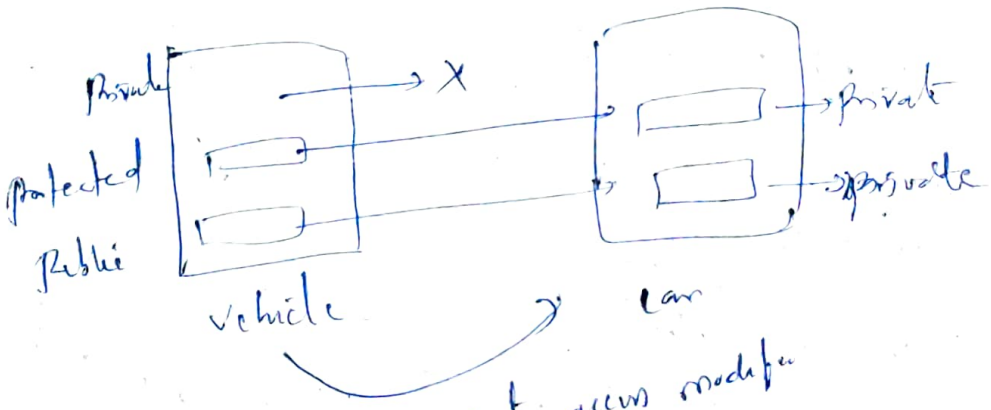
// These both can't be called outside the class as they are protected & private respectively

```
Car C;
C.color = "Black";
C.numTyres = 4;   → Error (it is not allowed)
C.BumGears = 5;
```

(Because protected property can be used only in class & not outside)



* when we inherit a class using protected modifier we can the protected members and public members of the base class becomes protected in child class



- when we inherit a class using private modifier. The protected members & public members of the base class becomes private in child class

---

Note: If we don't specify access modifier at the time of inheritance, by default it will consider as private.